

Implementasi Algoritma *Greedy Randomized Adaptive Search Procedure* (GRASP) dan Formulasi Model *Dotted Board* pada Penyelesaian *Cutting Stock Problem* Bentuk Irregular

Sisca Octarina*

Jurusan Matematika
Fakultas MIPA
Universitas Sriwijaya
sisca_octarina@unsri.ac.id

Sugandi Yahdin

Jurusan Matematika
Fakultas MIPA
Universitas Sriwijaya

Belly Wardhani

Jurusan Matematika
Fakultas MIPA
Universitas Sriwijaya

Abstrak—Cutting Stock Problem (CSP) merupakan masalah pemotongan bahan baku (*stock*) menjadi barang-barang (*item*) sesuai permintaan konsumen dengan aturan pemotongan tertentu. Penelitian ini menggunakan data penelitian Toledo *et al.* (2013) berupa 7 tipe *item* yang berbentuk tidak beraturan (*irregular*). Algoritma yang digunakan yaitu *Greedy Randomized Adaptive Search Procedure* (GRASP) untuk menentukan pola pemotongan yang diformulasikan ke dalam model *Dotted Board*. Berdasarkan hasil pembahasan, algoritma GRASP menghasilkan pola pemotongan yang optimal yang selanjutnya pola pemotongan tersebut diformulasikan ke dalam model *Dotted Board*. Solusi optimal dari model *Dotted Board* pada penelitian ini yaitu diperoleh jumlah *stock* minimum sebanyak 12 lembar *stock* yang digunakan untuk memenuhi permintaan konsumen.

Kata Kunci :—Irregular, Greedy Randomized Adaptive Search Procedure, Dotted Board.

I. PENDAHULUAN

Berkembang pesatnya dunia industri menimbulkan persaingan yang semakin kompleks antar perusahaan untuk memperoleh keuntungan. Persoalan klasik yang dihadapi adalah mencari cara untuk mengoptimalkan keuntungan. Permasalahan yang sering muncul dalam industri kertas adalah masalah pemotongan atau lebih dikenal *Cutting Stock Problem* (CSP). CSP merupakan masalah pemotongan bahan baku (*stock*) menjadi barang-barang (*item*) sesuai kebutuhan konsumen dengan aturan pemotongan tertentu. Menurut Suliman (2006) 2D-CSP dapat diklasifikasikan ke dalam *item* berbentuk beraturan (*regular*) dan tidak beraturan (*irregular*) [3]. Tujuan dari CSP adalah meminimumkan sisa pemotongan (*trim loss*) guna

mengurangi biaya bahan baku sehingga keuntungan dapat dioptimalkan.

Cutting Stock Problem juga bertujuan untuk memaksimalkan jumlah potongan *item* yang akan diproduksi sesuai permintaan konsumen. Algoritma yang digunakan untuk menyelesaikan CSP yaitu algoritma *heuristic* dan *meta-heuristic*. Algoritma *heuristic* merupakan algoritma paling umum yang digunakan namun seringkali menghasilkan solusi yang tidak optimal sehingga digunakan algoritma *meta-heuristic*, dimana hasilnya bukan solusi optimal lokal seperti pada algoritma *heuristic* [1]. Salah satu algoritma *meta-heuristic* yang biasa digunakan dalam penyelesaian CSP yaitu algoritma *Greedy Randomized Adaptive Search Procedure* (GRASP). Algoritma GRASP menggunakan dua tahap dalam proses penyelesaiannya, yaitu tahap konstruksi dan tahap pencarian solusi lokal. Octarina *et al.* (2018) berhasil mengimplementasikan algoritma GRASP dalam menyelesaikan 2D-CSP [2]. Algoritma GRASP mudah diimplementasikan dan memiliki kompleksitas waktu yang paling kecil dibandingkan algoritma lainnya. Algoritma GRASP juga dapat menyelesaikan CSP dengan memberikan solusi optimal dalam waktu yang tepat [4]. Selama ini algoritma GRASP hanya digunakan pada pembentukan pola pemotongan CSP untuk *item* berbentuk *regular*.

Pola pemotongan hasil dari implementasi algoritma GRASP dapat diformulasikan ke dalam model *Dotted Board*. Berdasarkan latar belakang tersebut, maka dilakukan pengimplementasian algoritma GRASP untuk pembentukan pola pemotongan *item* yang berbentuk *irregular*.

Selanjutnya pola-pola tersebut diformulasikan ke dalam model *Dotted Board* yang akan dicari pola pemotongan optimalnya sehingga fungsi tujuan dapat tercapai. Penelitian ini menggunakan data penelitian Toledo *et al.* (2013) karena merupakan data 2D-CSP yang terdiri dari *stock* berbentuk persegi panjang (*rectangular*) dan 7 tipe *item* berbentuk *irregular*.

II. TINJAUAN PUSTAKA

Pustaka-pustaka yang diperlukan dalam penelitian antara lain *Integer Linear Programming*, *Cutting Stock Problem*, algoritma *Greedy Randomized Adaptive Search Procedure*, dan model *Dotted-Board*.

2.1. Integer Linear Programming (ILP)

Bentuk umum model ILP dapat dilihat pada Model (1) sebagai berikut:

Fungsi tujuan:

Minimumkan atau maksimumkan

$$z = \mathbf{c}^T \mathbf{x} \quad (1)$$

dengan kendala:

$$A\mathbf{x}(\leq, =, \geq)\mathbf{v}$$

$$\mathbf{x} \geq \mathbf{0}, \mathbf{x} \in \mathbb{Z}^m$$

dimana:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix}, \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}, \mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{bmatrix}, \text{ dan } \mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

Keterangan:

z adalah fungsi tujuan yang dicari nilai optimumnya

A adalah matriks berukuran $n \times m$

\mathbf{c} adalah vektor koefisien fungsi tujuan yang memiliki m komponen

\mathbf{x} adalah variabel keputusan berbentuk vektor dengan m komponen

\mathbf{v} adalah vektor yang memiliki n komponen berisi jumlah permintaan

2.2. Cutting Stock Problem (CSP)

Cutting Stock Problem merupakan masalah optimasi dalam pemilihan alternatif pemotongan yang sering ditemukan dalam dunia industri seperti kertas, kayu, kaca, baja dan lain-lain. CSP pertama kali dikenalkan oleh Leonid Kantorovich yang merupakan ilmuwan asal Rusia pada tahun 1939, dan pertama kali berhasil diformulasikan oleh Kantorovich pada tahun 1960. Kemudian pada

tahun 1961 dan 1963 Gilmory dan Gomory berhasil memformulasikan CSP dalam model LP.

2.3. Algoritma *Greedy Randomized Adaptive Search Procedure* (GRASP)

Stock yang berukuran L panjang dan W lebar (L, W) dipotong sesuai *item* yang berukuran (l_i, w_i) , dimana $i = 1, 2, \dots, n$. Jumlah pemotongan setiap *item* i dipotong dengan batas P_i dan Q_i dengan $0 \leq P_i \leq Q_i$, dimana P_i dan Q_i merupakan batas bawah dan batas atas untuk setiap *item* i .

Berdasarkan nilai efisiensi setiap *item* i (e_i), tipe permasalahan dibedakan menjadi :

1. *Unweighted*, dimana $\forall i, e_i = 1$. Ukuran setiap *item* i sama dengan ukuran pada *stock*.
2. *Weighted*, dimana $\forall i, e_i \neq 1$. Beberapa ukuran *item* i tidak sama dengan ukuran pada *stock*.

Nilai efisiensi setiap *item* i dapat dicari dengan menggunakan Persamaan (2) :

$$e_i = \frac{v_i}{L_i} \quad (2)$$

(2.1)

dimana:

$$L_i = l_i w_i \quad (3)$$

Secara umum algoritma GRASP menurut Veldes *et al.* (2005) [6] adalah sebagai berikut:

1. Langkah 0 (Inisialisasi)
 \mathcal{L} sebagai *stock* yang akan dipotong berdasarkan himpunan *item* i .
 $\mathcal{P} = \{p_1, p_2, \dots, p_n\}$, himpunan *item* i yang akan dipotong.
 $\mathcal{C} = \emptyset$, himpunan *item* i yang dipotong.
2. Langkah 1 (Memilih persegi panjang pada \mathcal{L})
Mengambil \mathcal{L}^* yang merupakan persegi panjang terkecil pada \mathcal{L} dimana dapat memuat *item* pada \mathcal{P} . Jika \mathcal{L}^* tidak ada, maka berhenti. Sebaliknya, lanjutkan ke Langkah 2.
3. Langkah 2 (Memilih *item* i yang dipotong)
 - a. Memilih *item* p_i dengan batasan $n_i \leq Q_i$, *item* p_i yang dipilih selanjutnya dibentuk menjadi blok b^* untuk dipotong pada \mathcal{L}^* .
 - b. Memilih posisi pada \mathcal{L}^* untuk memotong b^* . Biasanya blok b^* yang dipotong tidak memenuhi \mathcal{L}^* , sehingga pemotongan dipotong pada sudut \mathcal{L}^* yang dekat dengan sudut.
 - c. *Update* nilai \mathcal{P}, \mathcal{C} , dan Q_i . *Update* \mathcal{C} dengan tipe i dan n_i pada *item* yang dipotong. Membuat $Q_i = Q_i - n_i$, jika $Q_i = 0$ maka hapus *item* i pada \mathcal{P} .

- d. Memindahkan blok b^* terhadap sudut terdekat pada *stock*.
4. Langkah 3 (*Update L*)
Menggabungkan *stock* yang belum terpakai untuk memotong *item* baru dari \mathcal{P} . Sehingga *update L* yang baru, kemudian lanjutkan ke Langkah 1.
- Pemotongan \mathcal{P} terhadap *item* yang dipotong dapat dibedakan menjadi 3 kriteria, sebagai berikut:
- Pemotongan terhadap $P_i l_i w_i$, mengutamakan potongan yang harus dipotong.
 - Pemotongan terhadap e_i , dimana $\forall i, P_i = 0$.
 - Pemotongan terhadap $l_i w_i$, dimana $\forall i, e_i = 1$.

2.4. Model Dotted Board

Bentuk umum model *Dotted Board* untuk 2D-CSP dilihat pada Model (4) sebagai berikut:

Fungsi tujuan:

Minimumkan

$$z = (c \cdot g_x) + x_t^M, \delta_t^d \quad (4)$$

$\forall d \in IFP_t, \forall t \in T$

dengan kendala:

$$\begin{aligned} \sum_{d \in IFP_t} \delta_t^d &= q_t \quad \forall t \in T; \\ \delta_u^e + \delta_t^d &\leq 1 \\ \forall e \in NFP_{t,u}^d, \forall t, u \in T, \forall d \in IFP_t; \\ \delta_t^d, \delta_u^e &\in \{0,1\} \quad \forall d \in IFP_t, \forall t \in T; \\ z &\geq 0 \end{aligned}$$

dimana:

$$\delta_t^d = \begin{cases} 1 & \text{jika titik referensi dari item} \\ & \text{t ipe t diposisikan pada titik } d \\ 0 & \text{sebaliknya} \end{cases}$$

Keterangan:

- z adalah fungsi tujuan yang dicari nilai optimumnya
- c adalah kolom *board*
- r adalah baris *board*
- g_x adalah resolusi *grid* pada sumbu x
- x_t^M adalah jarak *horizontal* dari titik referensi ke ujung *item*
- δ_t^d adalah variabel keputusan biner yang didefinisikan tiap pasangan
- q_t adalah banyaknya *item* tipe t yang harus diposisikan
- d, e adalah tipe titik pada *board*
- t, u adalah tipe potongan *item*
- IFP adalah *inner fit polygon*
- NFP adalah *no fit polygon*

III. METODE PENELITIAN

Langkah-langkah yang dilakukan dalam penelitian ini adalah :

- Mendeskripsikan data. Data yang digunakan dalam penelitian ini adalah data penelitian Toledo *et al.* (2013). Data-data tersebut meliputi ukuran panjang dan lebar *stock*, ukuran tiap *item* yang berbentuk *irregular*, dan batas atas jumlah permintaan.
- Menentukan pola pemotongan menggunakan algoritma GRASP pada 2D-CSP berdasarkan Sub-bab 2.3.
- Memformulasikan model *Dotted Board* berdasarkan Sub-bab 2.4.
- Menyelesaikan model *Dotted Board* dengan menggunakan program LINDO.
- Menganalisis hasil akhir yang diperoleh.

IV. HASIL DAN PEMBAHASAN



4.1. Pendeskripsian Data






Data yang digunakan pada penelitian ini merupakan data penelitian Toledo *et al.* (2013) [5] dapat dilihat pada Tabel 1. Berdasarkan Tabel 1. terdapat 7 *item* yang dipesan. Ketujuh *item* tersebut memiliki bentuk yang berbeda satu sama lain, sehingga dapat dikatakan sebagai *item* berbentuk *irregular*. Jumlah permintaan paling banyak yaitu 36 buah dan jumlah permintaan paling sedikit yaitu 6 buah. Ukuran *item* terbesar yaitu 20 cm^2 dan ukuran *item* terkecil yaitu 4 cm^2 .

4.2. Menentukan Pola Pemotongan Menggunakan Algoritma Greedy Randomized Adaptive Search Procedure pada Cutting Stock Problem Dua Dimensi untuk Item Berbentuk Irregular

Stock dengan ukuran panjang ($L = 22 \text{ cm}$) dan lebar ($W = 15 \text{ cm}$) (22,15), dipotong sesuai *item* yang berukuran (l_i, w_i) , $i = 1, 2, 3, 4, 5, 6, 7$. Pemotongan pada setiap *item* dilakukan secara *oriented*, yaitu pemotongan antara panjang dan lebar tidak dapat dibalik. Berdasarkan nilai P_i dan Q_i , tipe permasalahan dalam penelitian ini yaitu *constrained* atau *berkendala*, $\forall i, P_i = 0$, sehingga pemotongan yang dilakukan pada penelitian ini dilakukan berdasarkan pemotongan terhadap e_i . Data yang digunakan pada algoritma GRASP dapat dilihat pada Tabel 2.

Tabel 1. Ukuran *Item* dan Jumlah Permintaan

No	Bentuk <i>Item</i>	Ukuran <i>Item</i> L_i (cm^2)	Jumlah Permintaan v_i (buah)
1		14	20
2		16	23

3		4	6
4		17,5	29
5		6	10
6		17	30
7		20	36

Sumber: Toledo *et al.* (2013)

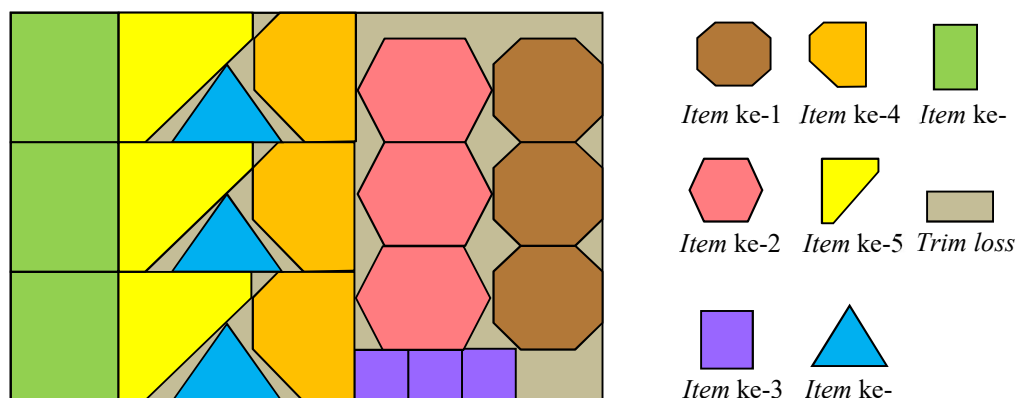
Tabel 2. Data yang Digunakan pada Algoritma GRASP

Item	L_i (cm ²)	P_i	Q_i	v_i	e_i
1	14	0	3	20	1,4286
2	16	0	3	23	1,4375
3	4	0	3	6	1,5
4	17,5	0	3	29	1,6572
5	6	0	3	10	1,6667
6	17	0	3	30	1,7647
7	20	0	3	36	1,8

Tabel 2. menunjukkan bahwa *item* pertama yang dipotong adalah *item* ke-7 karena mempunyai nilai efisiensi (e_7) terbesar yaitu 1,8, dengan luas

bangun sebesar 20 cm², batas bawah banyaknya pemotongan (P_7) adalah 0, batas atas banyaknya pemotongan (Q_7) adalah 3, dan jumlah permintaan (v_7) sebanyak 36, buah. *Item* selanjutnya yang akan dipotong adalah *item* ke-6, *item* ke-5, *item* ke-4, *item* ke-3, *item* ke-2, dan *item* ke-1. Pola pemotongan hasil dari implementasi algoritma GRASP dapat dilihat pada Gambar 1.

Berdasarkan Gambar 1. diperoleh banyaknya pemotongan pada *item* ke-1 yaitu $n_1 = 3$, maka $Q_1 = 3 - 3 = 0$. Selanjutnya meng-update nilai P, C , dan Q_1 , sehingga diperoleh $P = \emptyset$, $C = \{1\}$ dan $Q_1 = 0$. $P = \emptyset$ artinya tidak ada kumpulan *item* yang selanjutnya harus dipotong. $C = 1$ artinya *item* yang dipotong pada pola pemotongan ketujuh adalah *item* ke-1. $Q_1 = 0$ artinya tidak ada lagi *item* ke-1 yang harus dipotong.

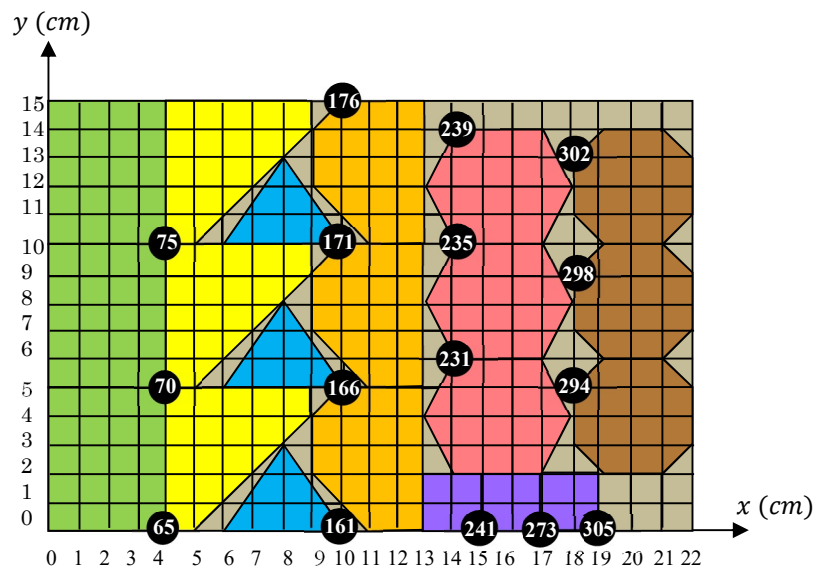


Gambar 1. Pola Pemotongan Optimal

Hasil pola pemotongan ketujuh menunjukkan nilai $P = \emptyset$ maka pemotongan dihentikan karena tidak ada lagi *item* yang harus dipotong. Pola pemotongan ketujuh merupakan pola pemotongan optimal karena semua batas atas jumlah permintaan tiap *item* telah terpenuhi yaitu 3 buah potongan untuk masing-masing tipe *item*.

Pola pemotongan optimal hasil implementasi algoritma GRASP pada Gambar 1. disajikan kembali dalam *Dotted Board* dapat dilihat pada Gambar 2.

4.3. Memformulasikan Model *Dotted Board*



Gambar 2. Pola Pemotongan Optimal pada *Dotted Board*

Berdasarkan Gambar 2. setiap *item* memiliki titik referensi yang diletakkan pada masing-masing sudut bangun tiap *item*. Titik referensi *item-item* dengan tipe ke-7 dan *item* tipe ke-6 diposisikan pada titik yang sama yaitu 65, 70 dan 75. Selanjutnya memformulasikan model *Dotted Board* berdasarkan Model 4, diperoleh model *Dotted Board* sebagai berikut:

$$\begin{aligned} \text{Minimumkan} \\ z = & \delta_7^{65} + 4\delta_7^{70} + 4\delta_7^{75} + 5\delta_6^{65} + 5\delta_6^{70} + 5\delta_6^{75} + 10\delta_5^{161} + \\ & 10\delta_5^{166} + 10\delta_5^{171} + 13\delta_4^{166} + 13\delta_4^{171} + 13\delta_4^{176} + \\ & 15\delta_3^{241} + 17\delta_3^{273} + 19\delta_3^{305} + 17\delta_2^{231} + 17\delta_2^{235} + \\ & 17\delta_2^{239} + 22\delta_1^{294} + 22\delta_1^{298} + 22\delta_1^{302} \end{aligned} \quad (5)$$

dengan kendala:

$$\begin{aligned} \delta_7^{65} + \delta_7^{70} + \delta_7^{75} &= 3 \\ \delta_6^{65} + \delta_6^{70} + \delta_6^{75} &= 3 \\ \delta_5^{161} + \delta_5^{166} + \delta_5^{171} &= 3 \\ \delta_4^{166} + \delta_4^{171} + \delta_4^{176} &= 3 \\ \delta_3^{241} + \delta_3^{273} + \delta_3^{305} &= 3 \\ \delta_2^{231} + \delta_2^{235} + \delta_2^{239} &= 3 \\ \delta_1^{294} + \delta_1^{298} + \delta_1^{302} &= 3 \end{aligned}$$




$$\begin{aligned} \delta_u^e + \delta_t^d &\leq 1 \\ \delta_t^d, \delta_u^e &\in \{0,1\} \\ z &\geq 0 \end{aligned}$$

Model (5) yang diperoleh selanjutnya diselesaikan dengan menggunakan program LINDO. Solusi optimal dari Model (5) dengan program LINDO adalah $\delta_7^{65} = 1, \delta_7^{70} = 1, \delta_7^{75} = 1, \delta_6^{65} = 1, \delta_6^{70} = 1, \delta_6^{75} = 1, \delta_5^{161} = 1, \delta_5^{166} = 1, \delta_5^{171} = 1, \delta_4^{166} = 1, \delta_4^{171} = 1, \delta_4^{176} = 1, \delta_3^{241} = 1, \delta_3^{273} = 1, \delta_3^{305} = 1, \delta_2^{231} = 1, \delta_2^{235} = 1, \delta_2^{239} = 1, \delta_1^{294} = 1, \delta_1^{298} = 1, \delta_1^{302} = 1$ dengan $z = 264$. Artinya diperoleh panjang minimum *stock* yang digunakan untuk memenuhi jumlah permintaan ketujuh *item* yang adalah **264 cm** atau setara dengan **12 lembar stock**, dimana panjang dan lebar *stock* masing-masing adalah **22 cm** dan **15 cm**.

Jumlah potongan yang dihasilkan untuk masing-masing tipe *item* berdasarkan solusi optimal dari Model (5) disajikan pada Tabel 3.

Tabel 3. Solusi Optimal Model *Dotted Board*

No	Bentuk <i>Item</i>	Ukuran <i>Item</i> L_i (cm ²)	Jumlah Potongan yang Dihasilkan (buah)	Jumlah Permintaan v_i (buah)	Jumlah Produk Berlebih (Surplus)
1		14	36	20	16
2		16	36	23	13
3		4	36	6	30
4		17,5	36	29	7

5		6	36	10	26
6		17	36	30	6
7		20	36	36	0

Berdasarkan Tabel 3. jumlah permintaan masing-masing tipe *item* terpenuhi, dan diperoleh potongan berlebih (*surplus*) yang dapat digunakan untuk memenuhi pemesanan berikutnya. Jumlah potongan *item* tipe ke-7 yang diperoleh adalah 36 buah dan tidak memiliki produk *surplus*, sedangkan *item* tipe ke-2 hingga *item* tipe ke-6 memiliki produk *surplus* dengan jumlah yang berbeda tiap *item*.

- Veldes, R. A., Parreño, F., and Tamarit, J. M. "A GRASP Algorithm for Constrained Two-Dimensional Non-Guillotine Cutting Problems", *Journal of the Operational Research Society*, 56, 2005, pp : 414 – 425.

V. KESIMPULAN

Berdasarkan hasil yang telah dicapai, dapat diambil beberapa kesimpulan yaitu:

- Implementasi algoritma GRASP dengan menggunakan data penelitian Toledo *et al.* (2013) pada penelitian ini menghasilkan pola pemotongan yang optimal, dimana semua jumlah batas atas permintaan ketujuh *item* berbentuk *irregular* terpenuhi.
- Solusi optimal model *Dotted Board* berdasarkan pola pemotongan hasil implementasi algoritma GRASP pada penelitian ini menghasilkan jumlah *stock* minimum sebanyak 12 lembar, dimana semua jumlah permintaan ketujuh *item* berbentuk *irregular* terpenuhi.

REFERENCES

- Karelahti, J., "Solving the Cutting Stock Problem in the Steel Industry", *Thesis*, Espoo : Department of Engineering Physics and Mathematics, Helsinki University of Technology, 2002, pp. 77.
- Octarina, S., Sholihatin, A., and Eliyati, N., "The Formulation of Gilmore and Gomory Model and Implementation of Greedy Randomized Adaptive Search Procedure (GRASP) Method in Cutting Stock Problem", *Submitted in Jurnal of Science and Technology Indonesia*, 2018.
- Suliman, S.M.A., "A Sequential Heuristic Procedure for the Two-Dimensional Cutting-Stock Problem", *International Journal of Production Economics*, 99(1-2), 2006, pp: 177 – 185.
- Tanadi, K., "Perbandingan Algoritma yang dipakai dalam 2D Knapsack Problem", *Makalah Strategi Algoritmik*, 2008.
- Toledo, F. M. B., Carravilla, M. A., and Ribeiro, C., Oliveira, J. F., and Gomes, A. M., "The Dotted Board Model: A New MIP Model for Nesting Irregular Shapes", *International Journal of Production Economics*, 142(2), 2013, pp: 478-487.